CS 271 - Project 0010

James Le

February 8, 2017

1. Prove Theorem 3.1:

   "For any two functions $f(n)$ and $g(n)$, $f(n) = \Theta(g(n))$ if and only if $f(n) = O(g(n))$ and $f(n) = \Omega(g(n))$."

   - Suppose $f(n) \in \Theta(g(n))$, then $\exists c_1, c_2, n_0, \forall n \geq n_0, 0 \leq c_1 g(n) \leq f(n) \leq c_2 g(n)$, if we just look at these inequalities separately, we have $c_1 g(n) \leq f(n)$ (because $f(n) \in \Omega(g(n))$) and $f(n) \leq c_2 g(n)$ (because $f(n) \in O(g(n))$).
   - Suppose that we had $\exists n_1, c_1, \forall n \geq n_1, c_1 g(n) \leq f(n)$ and $\exists n_2, c_2, \forall n \geq n_2, f(n) \leq c_2 g(n)$. Putting these together, and letting $n_0 = max(n_1, n_2)$, we have $\forall n \geq n_0, c_1 g(n) \leq f(n) \leq c_2 g(n)$, which means that $f(n) = \Theta(g(n))$.

2. Prove the following using the definitions of O, Ω, and Θ.

   (a) $n^2 + 3n - 20 = O(n^2)$

   According to the definition of O, we need to show that there exists $c \in \mathbb{R}^+$ and $n_0 \in \mathbb{N}$ such that

   $$0 \leq n^2 + 3n - 20 \leq cn^2 | \forall n \geq n_0$$

   Let $c = 2$ and $n = 4$. We then have $n^2 + 3n - 20 = 4^2 + 3 * 4 - 20 = 8 > 0$. For all values of $n > 4$, $n^2 + 3n - 20 \leq 2n^2$ since $3n - 20$ grows more slowly than n^2.

   (b) $n - 2 = \Omega(n)$

   According to the definition of Ω, we need to show that there exists $c \in \mathbb{R}^+$ and $n_0 \in \mathbb{N}$ such that

   $$0 \leq cn \leq n - 2 | \forall n \geq n_0$$

   Let $c = 0$ and $n = 2$. We then have $n - 2 = 2 - 2 = 0$. For all values of $n > 2$, $cn \leq n - 2$ since $cn = 0$ for $c = 0$ and $n - 2 > 0$.

   (c) $\log_{10} n + 4 = \Theta(\log_2 n)$

   According to the definition of Θ, we need to show that there exists $c_1, c_2 \in \mathbb{R}^+$ and $n_0 \in \mathbb{N}$ such that

   $$0 \leq c_1 \log_2 n \leq \log_{10} n + 4 \leq c_2 \log_2 n | \forall n \geq n_0$$

   Let $n = 2$, then we have $\log_2 n = \log_2 2 = 1$ and $\log_{10} n + 4 = \log_{10} 2 + 4 = 4.301$. To satisfy the Θ conditions, then we need $2c_1 \leq 4.301 \leq 2c_2$ for $n = 2$. We can arbitrarily pick any $c_1 \leq 2$ and $c_2 \geq 2.15$

   (d) $2^{n+1} = O(2^n)$

   According to the definition of O, we need to show that there exists $c \in \mathbb{R}^+$ and $n_0 \in \mathbb{N}$ such that

   $$0 \leq 2^{n+1} \leq c2^n | \forall n \geq n_0$$

   Let $c = 3$ and $n = 0$: we have then $0 \leq 2 \leq 3 * 2 = 6$. For all values of $n > 0$, $2^{n+1} \leq 3 * 2^n$ since 2^{n+1} grows more slowly than $3 * 2^n$.

(e) $\ln n = \Theta(\log_2 n)$

According to the definition of Θ, we need to show that there exists $c_1, c_2 \in \mathbb{R}^+$ and $n_0 \in \mathbb{N}$ such that

$$0 \leq c_1 \log 2n \leq lnn \leq c_2 \log 2n | \forall n \geq n_0$$

Let $n = 2$: then $\log_2 n = \log_2 2 = 1$ and $ln(n) = ln(2) = 0.69$. We then have $2c_1 \leq 0.69 \leq 2c_2$. We can arbitrarily pick any $c_1 \leq 0.34$ and $c_2 \geq 0.35$.

(f) $n^\epsilon = \Omega(\log_2 n)$ for any $\epsilon > 0$

According to the definition of Ω, we need to show that there exists $c_1, c_2 \in \mathbb{R}^+$ and $n_0 \in \mathbb{N}$ such that

$$0 \leq c \log n \leq n^\epsilon | \forall n \geq n_0$$

Let $n = 2$ and $c = 1$: we then have $c \log n = \log 2 = 1$ while $n^\epsilon = 2^\epsilon$. As ϵ gets arbitrarily smaller, 2^ϵ will approach 1, but never gets smaller than 1.

3. For each of the following recurrences, find a tight upper bound for $T(n)$. It will be useful to remember that

$$\sum_{i=0}^{\infty} a^i = \frac{1}{1-a}, if |a| < 1.$$

Prove that each is correct using induction. In each case, assume that $T(n)$ is constant for $n \leq 2$.

(a) $T(n) = 2T(n/2) + n^3$

Substitution:

$$T(n) = 2[2T(\frac{n}{4}) + (\frac{n}{2})^3] + n^3$$

$$T(n) = 4T(\frac{n}{4}) + (\frac{1}{4})n^3 + n^3$$

$$T(n) = 4[2T(\frac{n}{8}) + (\frac{n}{4})^3] + (\frac{1}{4})n^3 + n^3$$

$$T(n) = 8T(\frac{n}{8}) + (\frac{1}{16})n^3 + (\frac{1}{4})n^3 + n^3$$

The general formula is:

$$T(n) = (2^i)T(\frac{n}{2^i}) + n^3(1 + \frac{1}{4} + \frac{1}{16} + ... + \frac{1}{4^{i-1}})$$

$$T(n) = (2^i)T(\frac{n}{2^i}) + n^3 \sum_{j=0}^{i-1} (\frac{1}{4})^j$$

$$T(n) = (2^i)T(\frac{n}{2^i}) + n^3 \frac{1}{1 - 1/4}$$

$$T(n) = (2^i)T(\frac{n}{2^i}) + (\frac{4}{3})n^3$$

The last case is at $T(1)$: $n = 2i => i = \log_2 n$

$$T(n) = 2^{\log_2 n} T(1) + \frac{4}{3} n^3$$

$$T(n) = an + \frac{4}{3}n^3$$

<u>**Proof by Induction:**</u>

- **Base Case (n = 1):** $T(1) = a + \frac{4}{3} = a$
- **Inductive Hypothesis:**
  Assume that $T(k) = ak + \frac{4}{3}k^3$ for all $k = 1, 2, ..., n-1$
- **Induction Step:**

$$T(n) = 2T(\frac{n}{2}) + n^3$$

$$T(n) = 2[a\frac{n}{2} + \frac{4}{3}(\frac{n}{2})^2] + n^3$$

$$T(n) = an + \frac{1}{3}n^3 + n^3$$

$$T(n) = an + \frac{4}{3}n^3$$

(b) $T(n) = T(9n/10) + n$

<u>**Substitution:**</u>

$$T(n) = [T((\frac{9}{10})^2 n) + \frac{9}{10}n] + n$$

$$T(n) = T((\frac{9}{10})^3 n) + n(1 + \frac{9}{10} + \frac{9^2}{10^2})$$

The general formula is:

$$T(n) = T((\frac{9}{10})^i n) + n(1 + \frac{9}{10} + ... + \frac{9^{i-1}}{10^{i-1}})$$

$$T(n) = T((\frac{9}{10})^i n) + n\sum_{j=0}^{i-1}(\frac{9}{10})^j$$

$$T(n) = T((\frac{9}{10})^i n) + n\frac{1}{1-9/10}$$

$$T(n) = T((\frac{9}{10})^i n) + 10n$$

The last case is at $T(1)$: $\frac{9^i}{10^i} = \frac{1}{n} => n = (\frac{10}{9})^i => i = \log_{\frac{10}{9}} n$

$$T(n) = T(1) + 10n = a + 10n$$

<u>**Proof by Induction:**</u>

- **Base Case (n = 1):** $T(1) = a + 10 = a$
- **Inductive Hypothesis:**
  Assume that $T(k) = a + 10k$ for all $k = 1, 2, ..., n-1$

- **Induction Step:**

$$T(n) = T(\frac{9}{10}n) + n$$

$$T(n) = a + 10\frac{9}{10}n + n$$

$$T(n) = a + 10n$$

(c) $T(n) = 7T(n/3) + n^2$
Substitution:

$$T(n) = 7[7T(\frac{n}{9}) + (\frac{n}{3})^2] + n^2$$

$$T(n) = 49T(\frac{n}{9}) + \frac{1}{9}n^2 + n^2$$

$$T(n) = 49[7T(\frac{n}{27}) + (\frac{n}{9})^2] + \frac{1}{9}n^2 + n^2$$

$$T(n) = 343T(\frac{n}{27}) + n^2(1 + \frac{1}{9} + \frac{1}{81})$$

The general formula is:

$$T(n) = 7^iT(\frac{n}{3^i}) + n^2\sum_{j=0}^{i-1}(\frac{1}{9})^j$$

$$T(n) = 7^iT(\frac{n}{3^i}) + n^2\frac{1}{1 - 1/9}$$

$$T(n) = 7^iT(\frac{n}{3^i}) + \frac{9}{8}n^2$$

The last case is at $T(1)$: $\frac{n}{3^i} = 1 => n = 3^i => i = \log_3 n$

$$T(n) = 7^{\log_3 n}T(1) + \frac{9}{8}n^2$$

$$T(n) = 7^{\log_3 n}a + \frac{9}{8}n^2$$

Proof by Induction:

- **Base Case (n = 1):**
  $T(1) = 7^{\log_3 1}a + \frac{9}{8} = a + \frac{9}{8} = a$
- **Inductive Hypothesis:**
  Assume that $T(k) = 7^{\log_3 k}a + \frac{9}{8}k^2$ for all $k = 1, 2, ..., n-1$
- **Induction Step:**

$$T(n) = 7T(n/3) + n^2$$

$$T(n) = 7[7^{\log_3 \frac{n}{3}}a + \frac{9}{8}(\frac{n}{3})^2] + n^2$$

$$T(n) = 7 * 7^{\log_3 n - \log_3 3}a + \frac{7}{8}n^2 + n^2$$

$$T(n) = 7 * \frac{7^{\log_3 n}}{7^1} a + \frac{9}{8} n^2$$

$$T(n) = 7^{\log_3 n} a + \frac{9}{8} n^2$$

(d) $T(n) = T(\sqrt[2]{n}) + 1$

Substitution:

$$T(n) = T(n^{1/2}) + 1$$

$$T(n) = [T(n^{1/4}) + 1] + 1$$

$$T(n) = [T(n^{1/8}) + 1] + 1 + 1$$

The general formula is:

$$T(n) = T(n^{1/2^k}) + k$$

The last case is at $T(2)$:

$n^{1/2^k} = 2 => \log_2 n^{1/2^k} = \log_2 2 => \frac{1}{2^k} \log_2 n = 1 => \log_2 n = 2^k => \log_2(\log_2 n) = k$

$$T(n) = T(2) + \log_2(\log_2 n) = a + \log_2(\log_2 n)$$

Proof by Induction:

- **Base Case (n = 1):**
  $T(1) = a + \log_2(\log_2 1) = a + \log_2 0 = a + \infty = a$
- **Inductive Hypothesis:**
  Assume that $T(k) = a + \log_2(\log_2 k)$ for all $k = 1, 2, ..., n-1$
- **Induction Step:**

$$T(n) = T(n^{1/2}) + 1$$

$$T(n) = a + \log_2(\log_2 n^{1/2}) + 1$$

$$T(n) = a + \log_2(\frac{1}{2} \log_2 n) + 1$$

$$T(n) = a + \log_2 \frac{1}{2} + \log_2(\log_2 n) + 1$$

$$T(n) = a - \log_2 2 + \log_2(\log_2 n) + 1$$

$$T(n) = a + \log_2(\log_2 n)$$

(e) $T(n) = T(n-1) + \log n$

Substitution:

$$T(n) = T(n-2) + \log(n-1) + \log n$$

$$T(n) = T(n-3) + \log(n-2) + \log(n-1) + \log n$$

The general formula is:

$$T(n) = T(n-k) + \log(n-k+1) + ... + \log n$$

$$T(n) = T(n-k) + \log(n(n-1)...(n-k+1))$$

The last case is at $T(0)$ when $n = k$:

$$T(n) = T(0) + \log n! = a + \log n!$$

Proof by Induction:

- **Base Case (n = 1):**
  $T(n) = a + \log 1 = a$
- **Inductive Hypothesis:**
  Assume that $T(k) = a + \log k!$ for all $k = 1, 2, ..., n-1$
- **Induction Step:**

$$T(n) = T(n-1) + \log n$$

$$T(n) = a + \log(n-1)! + \log n$$

$$T(n) = a + \log(n(n-1)!)$$

$$T(n) = a + \log n!$$

4. Carefully prove by induction that the i-th Fibonacci number satisfies the equality

$$F_i = \frac{\phi^i - \Phi^i}{\sqrt{5}}$$

where $\phi = (1+\sqrt{5})/2$ is the golden ratio and $\Phi = (1-\sqrt{5})/2$ is its conjugate.
Note that

$$\phi^2 = (\frac{1+\sqrt{5}}{2})^2 = \frac{3+\sqrt{5}}{2}, \Phi^2 = (\frac{1-\sqrt{5}}{2})^2 = \frac{3-\sqrt{5}}{2}$$

recall $F_0 = 0, F_i = F_{i-1} + F_{i-2}$ for $i \geq 2$.
Base case:

$$\frac{\phi^0 - \Phi^0}{\sqrt{5}} = \frac{1-1}{\sqrt{5}} = 0 = F_0$$

$$\frac{\phi^1 - \Phi^1}{\sqrt{5}} = \frac{\frac{1+\sqrt{5}}{2} - \frac{1-\sqrt{5}}{2}}{\sqrt{5}} = 1 = F_1$$

Induction Hypothesis: Assume

$$F_n = \frac{\phi^n - \Phi^n}{\sqrt{5}} for 0 \leq n \leq i$$

Induction Step: We want to show

$$F_{i+1} = \frac{\phi^{i+1} - \Phi^{i+1}}{\sqrt{5}}$$

Now, using the induction hypothesis,

$$F + i + 1 = F_i + F_{i-1} = \frac{\phi^i - \Phi^i}{\sqrt{5}} + \frac{\phi^{i-1} - \Phi^{i-1}}{\sqrt{5}}$$

Using the definition of ϕ and Φ:

$$F_{i+1} = \frac{1}{\sqrt{5}}(\frac{(1+\sqrt{5})^i - (1-\sqrt{5})^i}{2^i} + \frac{(1+\sqrt{5})^{i-1} - (1-\sqrt{5})^{i-1}}{2^i})$$

By factoring and rearranging the terms, we have

$$F_{i+1} = \frac{1}{\sqrt{5}}(\phi^{i-1}\phi^2 - \Phi^{i-1}\Phi^2) = \frac{1}{\sqrt{5}}(\phi^{i+1} - \Phi^{i+1})$$